

WHAT IS CLAIMED IS:

1. A method comprising:
 - generating a message from a source application on a first computer system,
 - 5 wherein the source application is configured to communicate using an application programming interface (API) to a distributed computing infrastructure, and wherein the source application is network-unaware;
 - sending the message from the source application to the distributed computing infrastructure using the API;
 - 10 the distributed computing infrastructure translating the message from an original format to a portable format on the first computer system, thereby generating a portable message, wherein the portable message comprises metadata which comprise identifying characteristics of the source application; and
 - sending the portable message from the first computer system to a second
 - 15 computer system using peer-to-peer message passing between the first computer system, the second computer system, and optionally one or more intermediary computer systems;
 - receiving the portable message at the second computer system; and
 - the distributed computing infrastructure routing the portable message to a target application on the second computer system based on the metadata, wherein the target
 - 20 application is configured to communicate using the API to the distributed computing infrastructure API, and wherein the target application is network-unaware.
2. The method of claim 1,
 - wherein the source application routes network activity through the API to the
 - 25 distributed computing infrastructure, and wherein the target application receives network activity through the API from the distributed computing infrastructure.
3. The method of claim 1,

wherein the source application and the target application comprise chat applications enabling a user of the first computer system and a user of the second computer system to communicate using text;

wherein the message from the source application comprises text to be displayed
5 by the target application on the second computer system.

4. The method of claim 1,

wherein the source application and the target application comprise shared whiteboard applications enabling a user of the first computer system and a user of the
10 second computer system to communicate using graphical data on a virtual shared whiteboard;

wherein the message from the source application comprises graphical data to be displayed by the target application on the second computer system.

15 5. The method of claim 1,

wherein the source application and the target application comprise collaborative applications enabling a user of the first computer system and a user of the second computer system to communicate using collaborative data;

wherein the message from the source application comprises collaborative data to
20 be displayed by the target application on the second computer system.

6. The method of claim 1, further comprising:

determining an application type of the target application based on the metadata of the portable message;

25 determining whether an existing instance of the application type of the target application is running on the second computer system;

wherein the routing the portable message to the target application comprises routing the portable message to the existing instance if the existing instance of the application type of the target application is running on the second computer system; and

wherein the routing the portable message to the target application comprises routing the portable message to a new instance of the target application if the existing instance of the application type of the target application is not running on the second computer system.

5

7. The method of claim 1, further comprising:
delivering a payload of the message to the target application.

8. The method of claim 1, further comprising:
10 the distributed computing infrastructure translating the portable message from the portable format to the original format on the second computer system.

9. The method of claim 1, further comprising:
invoking functionality of the second computer system in response to the message.

15

10. The method of claim 9,
wherein the invoking functionality on the second computer system comprises instructing the target application to take one or more actions.

20 11. The method of claim 1,
wherein the portable message is sent from the first computer system to the second computer system and one or more additional computer systems using multicast peer-to-peer messaging.

25 12. The method of claim 1,
wherein the portable message is sent from the first computer system to the second computer system and one or more additional computer systems using broadcast peer-to-peer messaging.

13. The method of claim 1,
wherein the portable format comprises XML, and wherein the portable messages
comprise XML messages.

5 14. A carrier medium comprising program instructions, wherein the program
instructions are computer-executable to implement:

generating a message from a source application on a first computer system,
wherein the source application is configured to communicate using an application
programming interface (API) to a distributed computing infrastructure, and wherein the
10 source application is network-unaware;

sending the message from the source application to the distributed computing
infrastructure using the API;

the distributed computing infrastructure translating the message from an original
format to a portable format on the first computer system, thereby generating a portable
15 message, wherein the portable message comprises metadata which comprise identifying
characteristics of the source application; and

sending the portable message from the first computer system to a second
computer system using peer-to-peer message passing between the first computer system,
the second computer system, and optionally one or more intermediary computer systems;

20 receiving the portable message at the second computer system; and

the distributed computing infrastructure routing the portable message to a target
application on the second computer system based on the metadata, wherein the target
application is configured to communicate using the API to the distributed computing
infrastructure API, and wherein the target application is network-unaware.

25

15. The carrier medium of claim 14,
wherein the source application routes network activity through the API to the
distributed computing infrastructure, and wherein the target application receives network
activity through the API from the distributed computing infrastructure.

16. The carrier medium of claim 14,
wherein the source application and the target application comprise chat
applications enabling a user of the first computer system and a user of the second
5 computer system to communicate using text;
wherein the message from the source application comprises text to be displayed
by the target application on the second computer system.

17. The carrier medium of claim 14,
10 wherein the source application and the target application comprise shared
whiteboard applications enabling a user of the first computer system and a user of the
second computer system to communicate using graphical data on a virtual shared
whiteboard;
wherein the message from the source application comprises graphical data to be
15 displayed by the target application on the second computer system.

18. The carrier medium of claim 14,
wherein the source application and the target application comprise collaborative
applications enabling a user of the first computer system and a user of the second
20 computer system to communicate using collaborative data;
wherein the message from the source application comprises collaborative data to
be displayed by the target application on the second computer system.

19. The carrier medium of claim 14, wherein the program instructions are further
25 computer-executable to implement:
determining an application type of the target application based on the metadata of
the portable message;
determining whether an existing instance of the application type of the target
application is running on the second computer system;

wherein the routing the portable message to the target application comprises routing the portable message to the existing instance if the existing instance of the application type of the target application is running on the second computer system; and

5 wherein the routing the portable message to the target application comprises routing the portable message to a new instance of the target application if the existing instance of the application type of the target application is not running on the second computer system.

20. The carrier medium of claim 14, wherein the program instructions are further
10 computer-executable to implement:

delivering a payload of the message to the target application.

21. The carrier medium of claim 13, wherein the program instructions are further computer-executable to implement:

15 the distributed computing infrastructure translating the portable message from the portable format to the original format on the second computer system.

22. The carrier medium of claim 13, wherein the program instructions are further computer-executable to implement:

20 invoking functionality of the second computer system in response to the message.

23. The carrier medium of claim 22,
wherein the invoking functionality on the second computer system comprises instructing the target application to take one or more actions.

25

24. The carrier medium of claim 14,
wherein the portable message is sent from the first computer system to the second computer system and one or more additional computer systems using multicast peer-to-peer messaging.

25. The carrier medium of claim 14,
wherein the portable message is sent from the first computer system to the second
computer system and one or more additional computer systems using broadcast peer-to-
5 peer messaging.

26. The carrier medium of claim 14,
wherein the portable format comprises XML, and wherein the portable messages
comprise XML messages.

10

27. A system comprising:
a first computer system comprising a first CPU and a first memory; and
a second computer system comprising a second CPU and a second memory;
wherein the first computer system and the second computer system are
15 communicatively coupled via a network;
wherein the first memory stores program instructions which are executable by the
first CPU to:

generate a message from a source application on the first computer system,
wherein the source application is configured to communicate using an application
20 programming interface (API) to a distributed computing infrastructure, and wherein the
source application is network-unaware;

send the message from the source application to the distributed computing
infrastructure using the API;

using the distributed computing infrastructure, translate the message from
25 an original format to a portable format on the first computer system, thereby generating a
portable message, wherein the portable message comprises metadata which comprise
identifying characteristics of the source application; and

send the portable message from the first computer system to the second
computer system using peer-to-peer message passing between the first computer system,

the second computer system, and optionally one or more intermediary computer systems;
and

wherein the second memory stores program instructions which are executable by
the second CPU to:

5 receive the portable message at the second computer system; and
 using the distributed computing infrastructure, rout the portable message
to a target application on the second computer system based on the metadata, wherein the
target application is configured to communicate using the API to the distributed
computing infrastructure API, and wherein the target application is network-unaware.

10

28. The system of claim 27,

 wherein the source application routes network activity through the API to the
distributed computing infrastructure, and wherein the target application receives network
activity through the API from the distributed computing infrastructure.

15

29. The system of claim 27,

 wherein the source application and the target application comprise chat
applications enabling a user of the first computer system and a user of the second
computer system to communicate using text;

20

 wherein the message from the source application comprises text to be displayed
by the target application on the second computer system.

30. The system of claim 27,

 wherein the source application and the target application comprise shared
25 whiteboard applications enabling a user of the first computer system and a user of the
second computer system to communicate using graphical data on a virtual shared
whiteboard;

 wherein the message from the source application comprises graphical data to be
displayed by the target application on the second computer system.

31. The system of claim 27,

wherein the source application and the target application comprise collaborative applications enabling a user of the first computer system and a user of the second computer system to communicate using collaborative data;

wherein the message from the source application comprises collaborative data to be displayed by the target application on the second computer system.

32. The system of claim 27,

wherein the program instructions are executable by the second CPU to:

determine an application type of the target application based on the metadata of the portable message;

determine whether an existing instance of the application type of the target application is running on the second computer system;

wherein the routing the portable message to the target application comprises routing the portable message to the existing instance if the existing instance of the application type of the target application is running on the second computer system; and

wherein the routing the portable message to the target application comprises routing the portable message to a new instance of the target application if the existing instance of the application type of the target application is not running on the second computer system.

33. The system of claim 27,

wherein the program instructions are executable by the second CPU to:

deliver a payload of the message to the target application.

34. The system of claim 27,

wherein the program instructions are executable by the second CPU to:

using the distributed computing infrastructure, translate the portable message from the portable format to the original format on the second computer system.

35. The system of claim 27,
5 wherein the program instructions are executable by the second CPU to:
invoke functionality of the second computer system in response to the message.
36. The system of claim 35,
wherein the invoking functionality on the second computer system comprises
10 instructing the target application to take one or more actions.
37. The system of claim 27,
wherein the portable message is sent from the first computer system to the second
computer system and one or more additional computer systems using multicast peer-to-
15 peer messaging.
38. The system of claim 27,
wherein the portable message is sent from the first computer system to the second
computer system and one or more additional computer systems using broadcast peer-to-
20 peer messaging.
39. The system of claim 27,
wherein the portable format comprises XML, and wherein the portable messages
comprise XML messages.
25